# Department of Computing

# CS 212: Object Oriented Programming

# Lab 05: Inheritance

# Date: 11-03-2024

# Time: 9:00pm- 12:00 pm

# Instructor: Mr. Jaudat Mamoon

**Lab Engineer: Engr. Masabah Bint E Islam**

**Lab Tasks**

Reference Book: Java How to Program, 10th Ed, Deitel & Deitel (Available on LMS)

**Task# 1:**

Create two classes, "Animal" and "Dog", where "Dog" extends "Animal".

The "Animal" class should have an instance variable called "name" and a method called "speak()" that prints "An animal makes a sound" to the console.

The "Dog" class should have an instance variable called "breed" and a method called "speak()" that prints "A dog barks" to the console.

Write a test program, create an instance of "Dog" called "myDog" and set its name to "Buddy" and breed to "Golden Retriever". Call the "speak()" method on "myDog".

**Task# 2:**

Design a class named Person and its two subclasses named Student and Employee. Make Faculty and Staff subclasses of Employee.

A person has a name, address, phone number, and email address. A student has a class status (freshman, sophomore, junior, or senior). Define the status as a constant. An employee has an office, salary, and date hired. A faculty member has office hours and a rank. A staff member has a title. Override the toString() method in each class to display the class name and the person’s name.

Draw the UML diagram for the classes and implement them. Write a test program that creates a Person, Student, Employee, Faculty, and Staff, and invokes their toString() methods.

**Task# 3:**

The following UML class diagram illustrates an inheritance relationship, wherein the classes Circle and Rectangle have been extended from the class GeometricObject.

Text

Description automatically generated with medium confidence

You’re required to implement the classes GeometricObject and Rectangle.

The Rectangle class contains:

* Two double data fields named width and height that specify the width and height of the rectangle. The default values are 1.0 for both width and height.
* A no-arg constructor that creates a default rectangle.
* A constructor that creates a rectangle with the specified width and height.
* A method named getArea() that returns the area of this rectangle.
* A method named getPerimeter() that returns the perimeter.
* A method named toString() that returns a string description for the rectangle.

The toString() method is implemented as follows:

return "Rectangle: width = " + width + " height = " + height;

Write a test program that prompts the user to enter width and height of the rectangle, a color, and a Boolean value to indicate whether the rectangle is filled. The program should create a Rectangle object and set the color and filled properties using the input. The program should display the area, perimeter, color, and true or false to indicate whether it is filled or not.

**Task# 4:**

Implement the Shape hierarchy shown in below figure. Each TwoDimensionalShape should contain method getArea to calculate the area of the two-dimensional shape. Each ThreeDimensionalShape should have methods getArea and getVolume to calculate the surface area and volume, respectively, of the three-dimensional shape.

Create a program that uses an array of Shape references to objects of each concrete class in the hierarchy. The program should print a text description of the object to which each array element refers. Also, in the loop that processes all the shapes in the array, determine whether each shape is a TwoDimensionalShape or a ThreeDimensionalShape. If it’s a TwoDimensionalShape, display its area. If it’s a ThreeDimensionalShape, display its area and volume.

